16-811 Math Fundamentals for Robotics, Fall 2015

Project Report

3D Scanning from Shadows

Yi Hua, Mengxin Li

Introduction

The technology, 3D scanning, is widely used, especially after the emergence of 3D printers. Current 3D scanners used in industry can be quite accurate, however, requires expensive hardware. Inspired by Jean-Yves Bouguet's PhD thesis [1], we implemented a 3D scanner which makes use of the shadows of a stick onto the object that is to be scanned. Equipment required by this 3D scanner is quite simple: a camera, a stick, a lamp, a pencil and a chessboard. In this report, we will explain the method used to achieve the 3D scanning form shadows as well as present the results of the scanner. Potential improvements will be discussed in this report as well.

Method and Intermediate Results

Method Overview

The general idea of the 3D scanner is illustrated in Fig.1. With a stick sweeping above the object to be scanned, there will be a line shadow of the stick projected onto the object and the table on which the object is put on. The camera will take a video of the object during the scanning process, therefore a sequence of frames of the object and the shadow can be obtained from the camera. By projecting the shadow line λ_h in the image to the 3D space and intersecting it with the horizontal plane, Π_h, we can obtain the shadow line, Λ_h, in 3D space. Afterwards, the position of each point of the object in the 3D space can be obtained by intersecting the shadow plane defined by the light source and the shadow line, Λ_h, with the line defined by the camera center, O_c, and the shadow pixel, p. Once having a dense estimation of the points on the object to be scanned, we are able to reconstruct the 3D shape of the object.

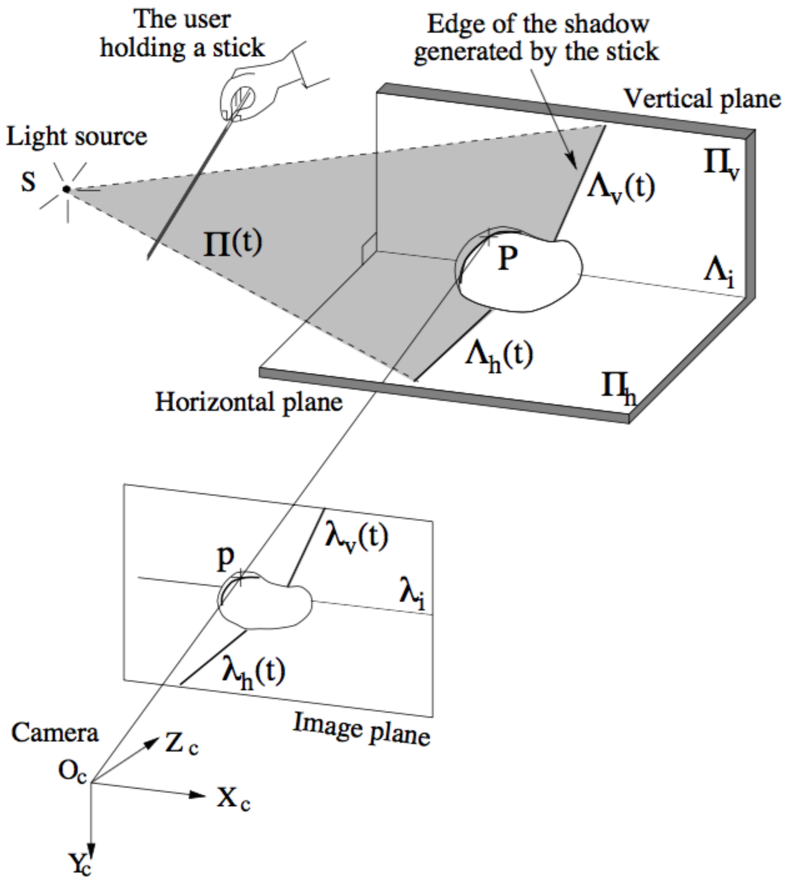

Fig.1 Method Overview

Camera Calibration

A camera can be modeled by intrinsic and extrinsic parameters which can be represented by the intrinsic and extrinsic matrix respectively. The intrinsic matrix is conventionally denoted by K, a 3x3 matrix defined by the focal length, principal point offset and skew parameter:

$$\mathrm{K} = \begin{bmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

The extrinsic matrix is a 3x4 matrix determined by the location and orientation of the camera center:

$$\mathrm{M} = [R \quad | \quad t] = [R \quad | \quad -Rc]$$

The projection matrix, P, which is the product of intrinsic and extrinsic matrix, project a certain point in the 3D world frame onto the 2D image plane:

$$\mathrm{P} = \mathrm{KM} = \mathrm{K}\,[R \quad | \quad t]$$

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \equiv P \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (1)$$

The aim of camera calibration is to obtain the intrinsic as well as extrinsic parameters of the camera. In this project, we used the method proposed by Tsai [2], which is to use a checkerboard with pre-known pattern and dimensions for camera calibration as illustrated in Fig.2. In our implementation, the *Camera Calibration Toolbox for MATLAB* [3] was used for this calibration purpose. The result of camera calibration is shown in Fig.3.

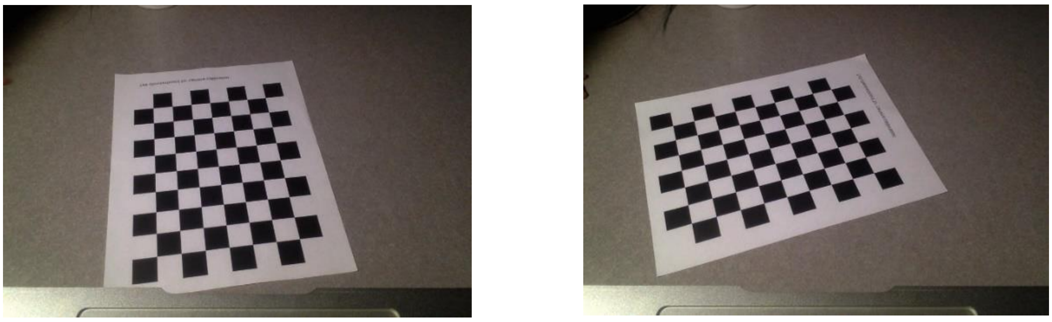

Fig.2 Camera Calibration Setup

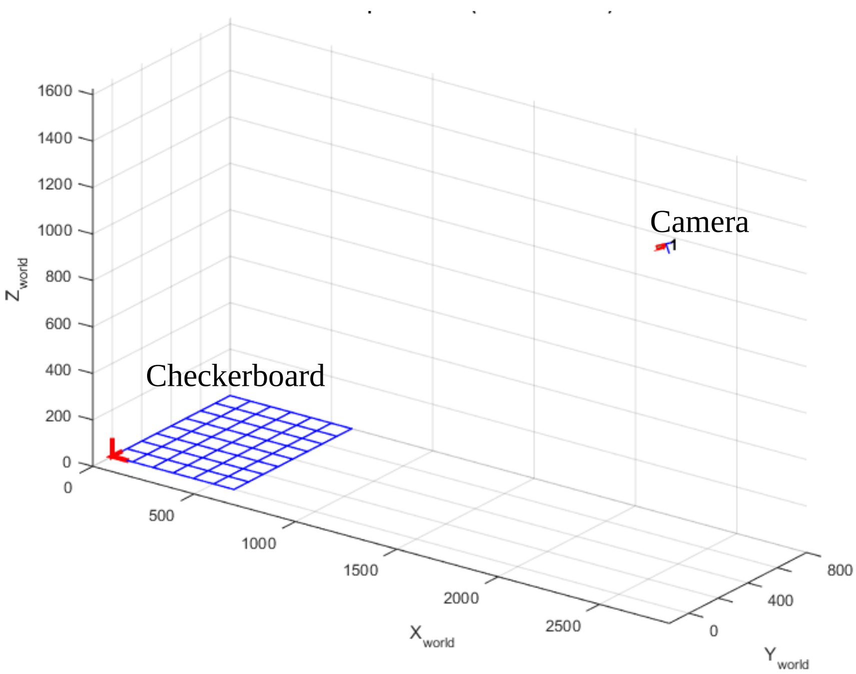

Fig.3 Camera Calibration Result (extrinsic)

Light Source Calibration

The purpose of light source calibration is to get the position of the light source point, for which the setup is shown in Fig.4. What we need for this calibration is a pencil standing on the table. As can be seen from Fig.5, the light source must be on the line Δ which is uniquely defined by the pencil top T and its shadow T_s. Since the height of the pencil can be pre-measured, the pencil top T can be derived from point B. And point B can be obtained by intersecting the line $O_c b$ and the horizontal plane, Π_h. Similarly, the position of point T_s can be obtained by intersecting the line $O_c t_s$ with the horizontal plane, Π_h. Then we move the pencil on the table while keeping the light source and camera fixed. In this way, we get multiple lines, Δ, and the light source position will be the intersection of these lines. A result of light source calibration is shown in Fig.6.

Fig.4 Light Source Calibration Setup

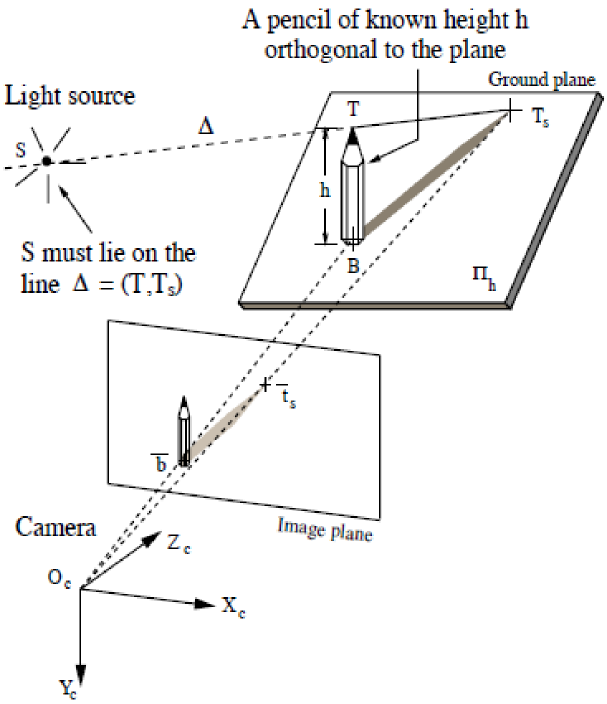

Fig. 5 Light Source Calibration

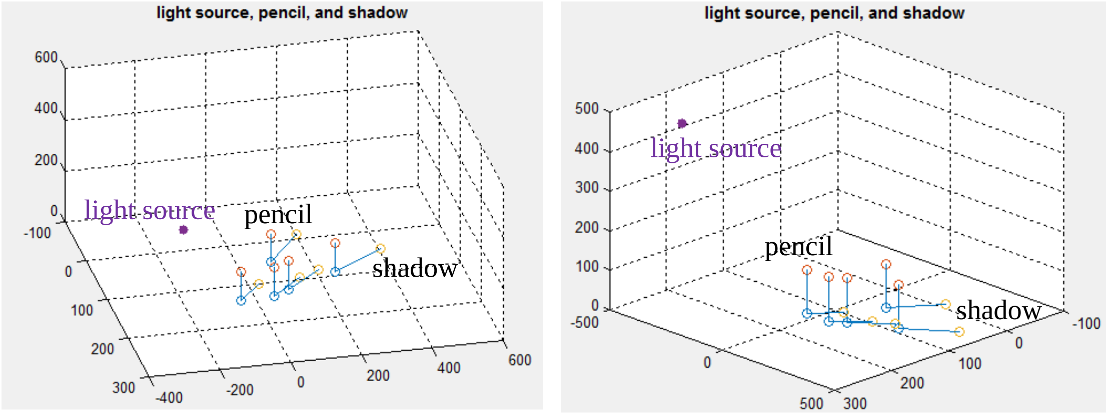

Fig.6 Light Calibration Result

Shadow Edge Detection, Shadow Line Fitting and Interpolation

In this step, we are going to localize the shadow edge in the frames. The property that the intensity of the shadow pixels changes drastically between two consecutive frames is used for shadow edge detection. To find these pixels in each frame, we define the difference image, $\Delta I(x, y, t)$ as follows:

$$I_{max}(x,y) \triangleq \max\{I(x, y, t)\}$$
$$I_{min}(x,y) \triangleq \min\{I(x, y, t)\}$$
$$\Delta I(x, y, t) \triangleq I(x, y, t) - \frac{1}{2}(I_{max}(x, y) + I_{min}(x, y))$$

Where I(x, y, t) denotes the frame sequence.

Then the shadow can be found by finding the pixels whose intensity in the difference image is less than 0 in the previous frame but greater than 0 in the current frame.

Since the stick is moving during the frame intervals, the shadow pixels detected using the method illustrated in the above paragraph will be a shadow band. However, what we need for shadow plane estimation (to be explained in the next section) is a line instead of a wide band. Therefore, the shadow edge indicated by the green line in Fig.7 needs to be localized from the shadow band that has been detected. And this can be done by selecting the left-most shadow pixels from the shadow band in each row of the image.

Once the shadow edge (the green line in Fig.7) is obtained, we can fit a line to the shadow edge. Two methods were tried in this project for line fitting. The first method is quite straight forward: Since two points uniquely define a line, we just pick the top and bottom points from the shadow edge, and the line that fits these two points will be the shadow line we are looking for. This method may suffer from errors caused by

outliers, however, works good in this project. The other method is to use all pixels on the shadow edge for line fitting using RANSAC, which is potentially more immunized to outliers than the first method. RANSAC is based on the assumption that the outliers cannot consistently vote for any single model while the inliers do. RANSAC is a probability based method for model fitting which does not guarantee a correct output, but just produce a reasonable result with a certain confidence which increase with the number of iterations. Compared with the first method, RANSAC requires much more computational power. Since the first method works adequately well in this project and saves computational power, we decided to use that method in our implementation. The result of line fitting is also shown in Fig.7 indicated by the yellow line.
/* interpolation */

Fig. 7 Spatial Edge Localization

Shadow Plane Estimation

As illustrated in Fig.1, the shadow plane is defined by the light source and the shadow line, Λ_h. The position of the light source has been obtained from light source calibration. The work remained is to get the shadow line, Λ_h, which can be acquired by project the 2D shadow line, λ_h, in the image back to the 3D space. The preimage of λ_h is a plane in the 3D space, and Λ_h is the intersection of the preimage with the horizontal plane, Π_h. Therefore, Λ_h can be obtained by finding out the intersection of two planes. However, to simplify the implementation, we can convert this problem into finding out the intersection of two lines with a plane: select two points on λ_h, then project these two points back into 3D space and get two lines. The intersection points of these two lines with Π_h uniquely determine Λ_h. To find these intersection points, the projection matrix of the camera is to be used, and since the horizontal plane Π_h has Z=0, according to equation (1), we can have the following:

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \equiv P \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} = \begin{bmatrix} p11 & p12 & p13 & p14 \\ p21 & p22 & p23 & p24 \\ p31 & p32 & p33 & p34 \end{bmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \equiv \begin{bmatrix} p11 & p12 & p14 \\ p21 & p22 & p24 \\ p31 & p32 & p34 \end{bmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \equiv \begin{bmatrix} p11 & p12 & p14 \\ p21 & p22 & p24 \\ p31 & p32 & p34 \end{bmatrix}^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \qquad (2)$$

Now with the position of the light source and two points on Λ_h, the shadow plane is uniquely determined for each frame. Below in Fig.8 a result of shadow plane estimation is shown.

/* Fig.8 shadow plane estimation result*/

Triangulation

Triangulation is the last step for the 3D reconstruction and it is the process to obtain the 3D position of a certain point with its 2D position in the image known. As discussed in the method overview section, the 3D position of such point can be obtained by intersecting the ray O_cp in Fig.1 with the shadow plane. In the previous step, we have finished the shadow plane estimation and defined it with the light source location, S (X_s, Y_s, Z_s) and the two points P_1 (X_1, Y_1, Z_1) and P_2 (X_2, Y_2, Z_2). To determine the line O_cp, 2 points are needed: the camera center O_c and another point on the preimage of pixel, p. The camera center has been obtained during the camera calibration process. The other point can be obtained by project the pixel, p, back to the 3D space, from which we get a line, and then intersect the line with the horizontal plane Π_h to obtain point P_0. This process is exactly the same as what we do to find out P_1 and P_2 using equation (2).

Now the problem becomes to find out the intersection of plane SP_1P_2, and line O_cP_0. Assume that the intersection P has the coordinate (X, Y, Z), then P must be in the plane SP_1P_2. Let matrix A to be the following:

$$A=\begin{bmatrix} X & Y & Z & 1 \\ X_S & Y_S & Z_S & 1 \\ X_1 & Y_1 & Z_1 & 1 \\ X_2 & Y_2 & Z_2 & 1 \end{bmatrix} \qquad (3)$$

Since points S, P_1, P_2 P are coplanar, rows of matrix A are linear dependent. Therefore, the determinant of A is 0:

$$\mathrm{Det}(A) = 0 \tag{4}$$

And since point P, O_c, P_0 are collinear, P (X, Y, Z) can be written as:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} X_C \\ Y_C \\ Z_C \end{pmatrix} + t\left(\begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} - \begin{pmatrix} X_C \\ Y_C \\ Z_C \end{pmatrix}\right) \tag{5}$$

Solving equations (3), (4) and (5) gives:

$$\mathrm{t} = -\frac{\det\begin{pmatrix} 1 & 1 & 1 & 1 \\ X_S & X_1 & X & X_C \\ Y_S & Y_1 & Y_2 & Y_C \\ Z_S & Z_1 & Z_2 & Z_C \end{pmatrix}}{det\begin{pmatrix} 1 & 1 & 1 & 0 \\ X_S & X_1 & X & X_0 - X_C \\ Y_S & Y_1 & Y_2 & Y_0 - Y_C \\ Z_S & Z_1 & Z_2 & Z_0 - Z_C \end{pmatrix}} \tag{6}$$

Then P (X, Y, Z) can be obtained by plugging the value of t in equation (6) back into equation (5). Using this method, 3D points of the object can be recovered from the corresponding 2D points in the frames. Then the 3D surface of the object can be recovered.

Overall Results

Fig.9 is a result of our 3D scanner that scans a hand. This result has achieved our expectation on this simple 3D scanner. As can be seen, the shape of the hand is adequately clear, and the 3D point cloud is quite dense.

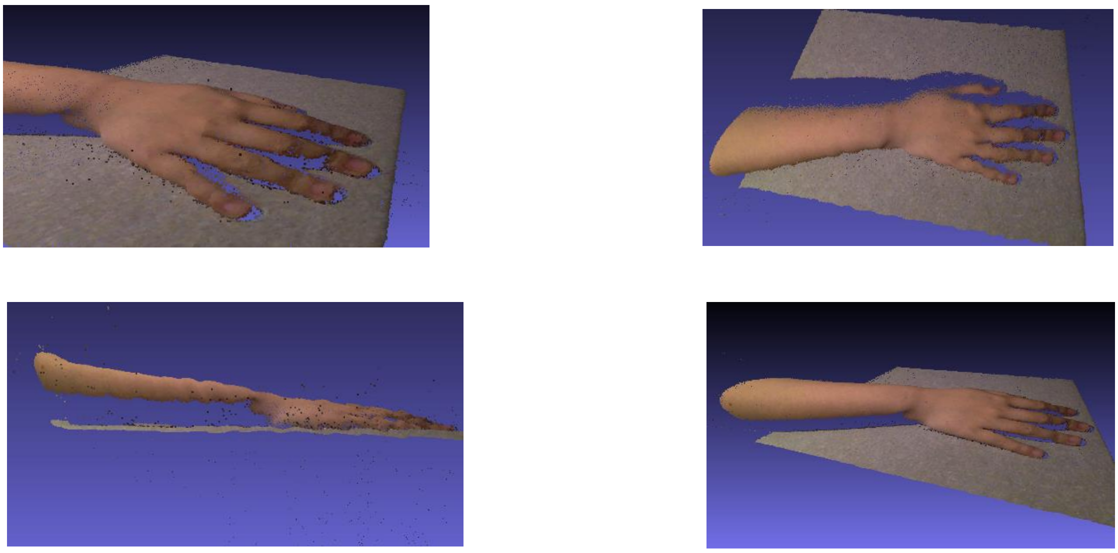

Fig.9 3D Scanning Result of a Hand

Conclusion

In this project, we have successfully implemented the 3D scanning from shadows. This demonstrated that a weak structured light system with simple and easy-to-access equipment can result in an adequate 3D scanner. Although the accuracy of this simple scanner is far less than the requirements on that for industrial use, it could be useful in the entertainment industry whose requirements on scanning accuracy would be lower.
Additionally, further improvements can be made to this scanner by merging multiple scan results. A single scan can only output the 3D surface that faces the camera, but multiple scans from different perspectives will give us the 3D reconstruction of surfaces seen from different angles, thus could output a full 3D reconstruction of an object.

Reference

[1] Jean-Yves Bouguet, 3D Photography on your desk, http://www.vision.caltech.edu/bouguetj/ICCV98/index.html
[2] Tsai R Y. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses[J]. Robotics and Automation, IEEE Journal of, 1987, 3(4): 323-344.
[3] Camera Calibration Toolbox for MATLAB, http://www.vision.caltech.edu/bouguetj/calib_doc